1. An In-Circuit Emulation system, comprising:

a microcontroller having a microcontroller clock;

a virtual microcontroller running in lock-step synchronization with the microcontroller;

a host computer running In-Circuit Emulation debug software, the host computer being in communication with the virtual microcontroller; and

a gatekeeper circuit coupled to the virtual microcontroller and the microcontroller, the gatekeeper circuit detecting when a watchdog timer expires in the microcontroller and notifying the host computer that the watchdog event has occurred.

2. The apparatus according to claim 1, further comprising a gatekeeper clock running independent of the microcontroller clock to clock operations carried out in the gatekeeper circuit.

3. The apparatus according to claim 2, wherein the gatekeeper circuit further comprises switching means for replacing the microcontroller clock with the gatekeeper clock for clocking the virtual microcontroller when a watchdog event occurs.

4. The apparatus according to claim 1, wherein the gatekeeper circuit further comprises means for asserting and holding a reset on the microcontroller when a watchdog event occurs.

5. The apparatus according to claim 1, wherein the gatekeeper circuit detects that a watchdog event has occurred by determining that the microcontroller clock is not active and that a data bus is in a prescribed logic state.

6. The apparatus according to claim 5, wherein the data bus comprises first and second data lines and wherein the prescribed logic state comprises both of the first and second data lines being held at a logic high state.

7. The apparatus according to claim 1, further comprising a gatekeeper clock running independent of the microcontroller clock to clock operations carried out in the gatekeeper circuit, and wherein the gatekeeper circuit further comprises switching means for replacing the microcontroller clock with the gatekeeper clock for clocking the virtual microcontroller when a watchdog event occurs.

8. The apparatus according to claim 7, wherein the gatekeeper circuit further comprises means for asserting and holding a reset on the microcontroller when a watchdog event occurs.

9. The apparatus according to claim 8, wherein the gatekeeper circuit detects that a watchdog event has occurred by determining that the microcontroller clock is not active and that a data bus is in a prescribed logic state.

10. The apparatus according to claim 9, wherein the data bus comprises first and second data lines and wherein the prescribed logic state comprises both of the first and second data lines being held at a logic high state.

11. A method of processing a watchdog timer event using a gatekeeper circuit, comprising:

determining that a watchdog timer event has occurred in a microcontroller, the microcontroller running in lock-step synchronization with a virtual microcontroller;

asserting a reset in the microcontroller and by holding the reset line of the microcontroller in an asserted state;

providing a gatekeeper clock signal to the virtual microcontroller;

notifying a host computer running In-Circuit Emulation software that a watchdog timer event has occurred; and

permitting the host computer to query memory locations and registers of the virtual microcontroller.

12. The method according to claim 11, wherein the determining further comprises determining that a microcontroller clock is not active and that a data bus is in a prescribed logic state.

13. The method according to claim 12, wherein the data bus comprises a two line data bus and wherein the prescribed logic state comprises the two data lines being in a logic high state.

14. The method according to claim 11, wherein the gatekeeper clock runs independently of the microcontroller clock to clock operations carried out in the gatekeeper circuit.

15. A method of processing a watchdog timer event using a gatekeeper circuit, comprising:

determining that a watchdog timer event has occurred in a microcontroller, the microcontroller running in lock-step synchronization with a virtual microcontroller;

asserting a reset in the microcontroller and by holding the reset line of the microcontroller in an asserted state;

disabling a microcontroller clock signal from the virtual microcontroller and replacing the microcontroller clock signal with a gatekeeper clock signal to the virtual microcontroller; and

notifying a host computer running In-Circuit Emulation software that a watchdog timer event has occurred.

16. The method according to claim 15, further comprising permitting the host computer to query memory locations and registers of the virtual microcontroller.

17. The method according to claim 15, wherein the determining further comprises determining that a microcontroller clock is not active and that a data bus is in a prescribed logic state.

18. The method according to claim 15, wherein the data bus comprises a two line data bus and wherein the prescribed logic state comprises the two data lines being in a logic high state.

19. The method according to claim 15, wherein the gatekeeper clock runs independently of the microcontroller clock to clock operations carried out in the gatekeeper circuit.